Journal of Engineering Science and Technology
EURECA 2014 Special Issue January (2015) 82 - 97

SOFT COMPUTING APPROACH TO DESIGN PID CONTROLLER FOR TANK LIQUID LEVEL CONTROL PROBLEM

K. SURESH MANIC[1,]*, SARATH ANANTHASIVAM[1], V. RAJINIKANTH[2]

[1]School of Engineering, Taylor's University, Taylor's Lakeside Campus,
No. 1 Jalan Taylor's, 47500, Subang Jaya, Selangor DE, Malaysia.
[2]Department of Electronics and Instrumentation Engineering,
St. Joseph's College of Engineering, Chennai 600 119, Tamilnadu, India.
*Corresponding Author: SureshManic.Kesavan@taylors.edu.my

Abstract

In this paper, an investigation has been made to design the best possible Proportional + Integral + Derivative (PID) controller for the Hopper type tank and Spherical tank system. In order to identify the optimal controller parameters, soft computing schemes, such as Particle Swarm Optimization (PSO), Bacterial Foraging Optimization (BFO), Firefly Algorithm (FA), and Cuckoo Search (CS) are considered. In the proposed work, minimization of a weighted sum of Objective Function (OF) is adopted to guide the soft computing technique based controller design procedure. The qualitative and quantitative analysis is carried out to validate the performance of the considered procedure. The results evident that, the Brownian walk guided algorithms offers better performance compared to the PSO.

Keywords: Hopper type tank, Spherical tank, PID controller, Brownian walk, Heuristic algorithms.

1. Introduction

Design of appropriate PID controller is widely preferred in process industries to enhance the quantity of final product without compromising the quality. In the control literature, a number of traditional [1-4] and soft computing approach [5-7] based PID design procedures are discussed by most of the researchers for stable, unstable and nonlinear systems. Even though there exists a number of traditional tuning procedures [1, 8], heuristic and metaheuristic algorithm based approaches are also considered by the researchers in recent years [9-11]. These soft computing approaches are preferred in control applications because of its

Nomenclatures

D	Dimension of search
$G_p(s)$	Process model
K	Process gain
K_d	Derivative gain
K_i	Integral gain
K_p	Proportional gain
M_p	Peak overshoot
t_s	Settling time

Greek Symbols

θ	Process time delay
τ	Process time constant

Abbreviations

BW	Brownian walk
FOPTD	First order plus time delay
ITAE	Integral time absolute error
ITSE	Integral time square error
PID	Proportional + Integral + Derivative
PSO	Particle swarm optimization

simplicity, optimization ability, and speed of response. Due to its flexibility; the soft computing approaches can easily adapt with existing classical controller design procedures. Hence, in recent days, it is used as a tool to design classical and modified structured controllers for a class of stable, unstable, and nonlinear process models

In this paper, soft computing based PID controller design procedure is discussed for hopper tank and spherical tank system and performance comparison is presented between algorithms, such as Particle Swarm Optimization (PSO), Bacterial Foraging Optimization (BFO), Firefly Algorithm (FA), and Cuckoo Search (CS) for the reference tracking and input disturbance operations.

The paper is organized as follows: Section 2 presents the problem formulation for PID controller design. Process description is discussed in Section 3. Section 4 presents the overview of the heuristic algorithms considered in this study and its implementation. Experimental results are evaluated and discussed in Section 5. Conclusion of the present research work is given in Section 6.

2. Problem Formulation

In order to improve the performance of closed loop systems, optimally tuned PID controller is required. Figure 1 depicts the illustration of a conventional closed loop control system. In this, the controller G_c(s) has to support closed loop stability, smooth reference tracking, and efficient disturbance rejection [2]. In this, G_c(s) is used to improve both the steady state as well as the transient response of G_p(s). The main objective here is to make Y(s) = R(s). In this framework, G_c(s) continuously adjusts the value of U_c(s) until the error E(s) is zero irrespective of D(s).

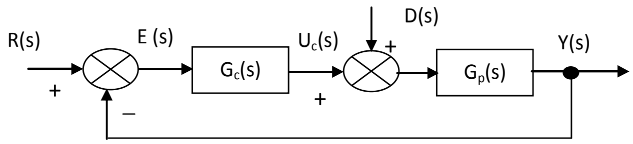

Fig. 1. General closed loop system structure.

The closed loop response of the system with reference R(s) and supply disturbance D(s) can be expressed as:

$$Y(s)=\left[\frac{G_p(s)G_C(s)}{1+G_p(s)G_C(s)}\right]R(s)+\left[\frac{G_p(s)}{1+G_p(s)G_C(s)}\right]D(s) \tag{1}$$

where the complementary sensitivity function and sensitivity function of the above loop is represented in Eqs. (2) and (3) respectively.

$$T(s)=\frac{Y(s)}{R(s)}=\left[\frac{G_p(s)G_C(s)}{1+G_p(s)G_C(s)}\right] \tag{2}$$

$$S(s)=\left[\frac{1}{1+G_p(s)G_C(s)}\right] \tag{3}$$

The final steady state response of the system for the set point tracking and the disturbance rejection is presented below

$$y_R(\infty)=\lim_{t\to\infty} s\ Y_R(s)=\lim_{t\to\infty} s\ x\left[\frac{G_p(s)G_C(s)}{1+G_p(s)G_C(s)}\right]\left(\frac{A}{s}\right)\ =\ A \tag{4}$$

$$y_D(\infty)=\lim_{t\to\infty}\ s\ x\left[\frac{G_p(s)}{1+G_p(s)G_C(s)}\right]\left(\frac{L}{s}\right)=0 \tag{5}$$

where A is amplitude of the reference signal and L is the disturbance amplitude.

To achieve a satisfactory $y_R(\infty)$ and $y_D(\infty)$, it is necessary to use optimally tuned PID parameter values. In this paper, non-interacting form of PID structure is adopted. In this structure, a low pass filter is available with the derivative term to minimize the effect of measurement noise.

The PID structure is defined below:

$$G_c(s)=K_p\left[1+\frac{1}{T_i\ s}+\frac{T_d\ s}{\frac{T_d s}{N_f}+1}\right] \tag{6}$$

where $K_p / T_i = K_i$, K_p x $T_d = K_d$ and N_f = filter constant = 10.

3. Process Description

Industrial systems are nonlinear in nature. Most of the real time process loops are characterized by an equivalent higher order modelling equations. These nonlinear processes can be efficiently modelled as linear processes based on operating regions. In the proposed work, the FOPTD model of the hopper tank systems and spherical tank systems are considered.

3.1. Hopper tank

Hopper type process model is shown in Fig. 2(a). This system has conical bottom portion and cylindrical upper portion [12]. In this work, we considered the conical portion alone. The conical portion is highly nonlinear in nature and developing the mathematical model based on the working region is a hard job.

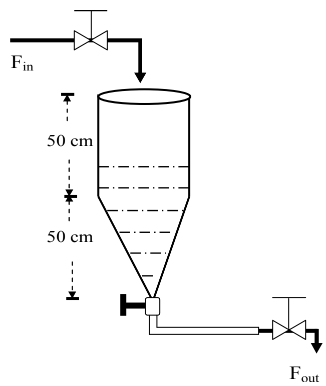

Fig. 2(a) Hopper Tank System.

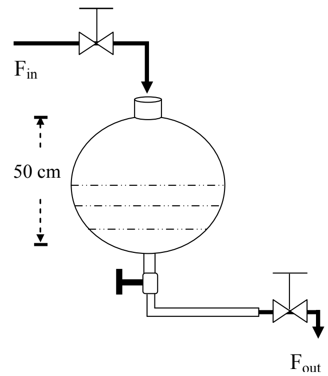

Fig. 2(b) Spherical Tank System.

In this paper, the mathematical model of the hopper tank system with various operating regions discussed by Kesavan et al. [12] is considered. In their work, the conical portion of the tank is considered and four stable FOPTD models are developed based on the different operating regions as shown in Table 1.

Table 1. Process parameters at different operating range.

Inflow %	Range (cm^3/s)	Level range (cm)	K	τ	θ
40	(I Region)	0 - 15	2.70	0.75	0.15
60	(II Region)	15 - 30	0.68	1.50	0.70
80	(III Region)	30 - 40	0.18	0.78	0.22
100	(IV Region)	40 - 50	0.09	0.30	0.30

3.2. Spherical tank

The spherical tank is the one of the nonlinear system widely discussed by the researchers in the literature [13, 14]. In this work, the spherical tank system shown in Fig. 2(b) is considered. The operating region of the spherical tank system is chosen as 18 cm. At this operating region, 29.549% of tank is filled with the water (19.34 L) and 70.451% is filled with the air, and the developed mathematical model around the corresponding operating range is presented below [14]

$$G(s) = \frac{3.6215\ e^{-11.7s}}{330.46s + 1} \tag{7}$$

In the proposed work, the conical tank portion with a height of 50 cm is considered for hopper tank system and the spherical tank with a diameter of 50 cm is considered to develop the required FOPTD model of the system around the operating regions.

4. Overview of Heuristic Algorithms Considered

In recent years, a considerable number of heuristic and meta-heuristic algorithms are proposed by the researchers to deal with variety of optimization problems. In this work, well known algorithms, such as Particle Swarm Optimization (PSO), Bacterial Foraging Optimization (BFO), Firefly Algorithm (FA) and Cuckoo Search (CS) are considered to design the PID controller.

4.1. Particle swarm optimization

PSO algorithm was proposed by Kennedy and Eberhart in 1995 [15]. Compared to other agent-based stochastic optimization techniques, PSO offers better exploration performance, faster and more stable convergence rates. Also, the number of initial algorithm parameters to be assigned is very few compared to other nature-inspired algorithms existing in the literature [13].

During the optimization search, each particle remembers its best position attained so far (i.e., pbest - $P_{i,D}^{t}$), and also obtains the global best position information achieved by any particle in the population (i.e., gbest - $G_{i,D}^{t}$).

At iteration t, each particle i has its position defined by $X_{i.n}^{t} = [X_{i,1}, X_{i,2}, ..., X_{i,D}]$ and velocity defined as $V_{i,n}^{t} = [V_{i,1}, V_{i,2}, ..., V_{i,D}]$ in search dimension D.

Velocity and position of each particle in the next iteration can be calculated as

$$V_{i,D}^{t+1} = W * V_{i,D}^{t} + C_1 * R_1 * (P_{i,D}^{t} - X_{i,D}^{t}) + C_2 * R_2 * (G_{i,D}^{t} - X_{i,D}^{t}) \tag{8}$$

$$X_{i,D}^{t+1} = X_{i,D}^{t} + V_{i,D}^{t+1} \tag{9}$$

where $i = 1, 2, \ldots, N$; $n = 1, 2, \ldots, D$; C_1 is the cognitive parameter (typically 2); C_2 is the social parameter (typically 2); R_1 and R_2 are random numbers in the range 0-1; and W is the weighting parameter (typically 0.75) [13].

4.2. Brownian walk guided algorithms

In the heuristic algorithm based search, generally the success towards the optimal solution mostly relies on the guiding procedure. Most of the recently developed nature inspired optimization search process is guided by Lévy Flight (LF) and Brownian Walk (BW) strategy. In this paper, BW is considered to guide algorithms, such as BFO, FA and CS

The Brownian Walk (BW) is a subdiffusive non-markovian process, which follows a Gaussian distribution with zero mean and time-dependent variance. In this work, the BW recently discussed by Raja et al. [16] is adopted.

The following formulae are considered

Brownian distribution = $B(s) = A.|s|^{\alpha/2}$ (10)

$$A = \beta\Gamma(\beta)\sin\left(\frac{\beta\pi}{2}\right)\frac{1}{\pi} \quad (11)$$

where A is the random variable, β is the spatial exponent, α is the temporal exponent, and $\Gamma(\beta)$ is a Gamma function.

A detailed description of BW is discussed in [16-18]. Figure 3 shows the exploration traces made by a BW guided single agent in a '*D*' dimensional search space. When iteration increases, all the agents in the algorithm will converge towards the optimal values of the controller parameters in the search universe.

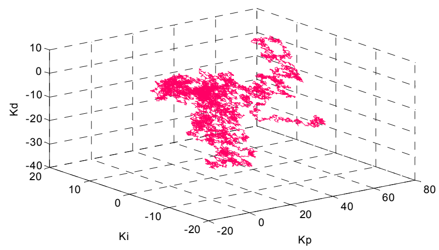

Fig. 3. Search traces made by a single agent in Brownian distribution.

4.2.1. Bacterial foraging optimization

BFO algorithm was developed in 2002 by mimicking the foraging behaviour of *E. coli* bacteria. In the proposed work, the enhanced BFO algorithm discussed in [14] is considered. The initial algorithm parameters are assigned as follows:

Number of *E.Coli* bacteria = N

$$N_c = \frac{N}{2};\ N_s = N_{re} \approx \frac{N}{3};\ N_{ed} \approx \frac{N}{4};\ N_r = \frac{N}{2};\ \text{Ped} = \left(\frac{N_{ed}}{N + N_r}\right);$$

$$d_{attractant} = W_{attractant} = \frac{N_S}{N}; \quad \text{and} \quad h_{repellant} = W_{repellent} = \frac{N_C}{N}. \quad (12)$$

In BFO algorithm, optimization accuracy and convergence rate depends on the chemotaxis operation and it can be expressed as in Eq. (12) [14].

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \quad (13)$$

where θ^i *(j, k, l)* shows i^{th} bacterium at j^{th} chemotactic, k^{th} reproductive and l^{th} elimination-dispersal step; *C(i)* is the step size in the random direction, and $\Delta(i)$ is a random vector of size {-1, 1}. In the proposed work, Eq. (12) is modified as shown below:

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \oplus B(s) \quad (14)$$

where the symbol ⊕ represents the entry wise multiplication, and *B(s)* is the Brownian walk operator.

4.2.2. Firefly algorithm

Firefly algorithm is a nature inspired metaheuristic algorithm initially proposed by Yang [19]. It is developed by replicating the flashing illumination patterns generated by the firefly. This bioluminescence with varied flashing patterns is used to establish communication between two neighbouring insects, to search for pray and also to find mates [20].

The overall performance (exploration time, speed of convergence, and optimization accuracy) of the FA depends on the OF, which monitors the search. For a minimization problem, luminance of a firefly is considered based on the following relation luminance = 1/OF. A detailed explanation about FA is available in literature [17-20].

In FA, the light intensity at a particular distance d from the light source X_i^t obeys the inverse square law. The light intensity of a firefly I, as the distance d increases interms of $I \propto 1/d^2$. The movement of the attracted firefly i towards a brighter firefly j can be determined by the following position update equation;

$$X_i^{t+1} = X_i^t + \beta_0 e^{-\gamma d_{ij}^2} (X_j^t - X_i^t) + \alpha.\mathrm{sign}(\mathrm{rand} - ½) \quad \oplus \quad \mathrm{B(s)} \qquad (15)$$

where X_i^{t+1} is updated position of firefly, X_i^t is initial position of firefly, $\beta_0 e^{-\gamma d_{ij}^2} (X_j^t - X_i^t)$ is the attraction between fireflies, and *B(s)* is the Brownian walk operator.

4.2.3. Cuckoo search technique

CS was initially proposed by Yang and Deb [19]. It is based on the breeding tricks of parasitic cuckoos. CS algorithm has the following assumption [17]

- Each cuckoo lays an egg and deposits in an arbitrarily chosen nest
- The nest with good survived egg will be approved over to the next generation. Cuckoo's egg normally hatches several days before than the host's eggs. The cuckoo chick grows quicker and expels the host's eggs.
- In a search universe, the number of host nest is fixed. The host bird discovers the cuckoo's egg with a probability $p_a \in [0,1]$. When the egg is discovered, host bird may eliminate it from nest, or simply abandon the nest and build a new nest.

In CS, during the optimization search, the new solution ($X_i^{(t+1)}$) mainly depends on the old solution ($X_i^{(t)}$) and the search guiding procedure.

In this work, the following expressions are considered to find the new solution

$$X_i^{(t+1)} = X_i^{(t)} + \alpha \oplus B(s) \qquad (16)$$

where $\alpha > 0$ is the succeeding step.

4.3. Implementation

Heuristic algorithm based PID controller tuning procedure is depicted in Fig. 4. For a given process model, the algorithm finds the optimal K_p, K_i and K_d values from the search universe '*D*' by minimizing the objective function. In the literature, a number of objective functions are existing to support heuristic algorithm based PID design task. In this work, OF is framed by considering the time domain constraints (M_p, and t_s) and the errors (ITAE, and ITSE) as presented below:

$$J_{min} = (w_1 \times M_p) + (w_2 \times t_s) + (w_3 \times ITAE) + (w_4 \times ITSE) \tag{17}$$

$$ITAE = \int_0^{500} t\left|e^2(t)\right|dt = \int_0^{500} t\left|[r(t) - y(t)]^2\right|dt \tag{18}$$

$$ITSE = \int_0^{500} t\, e^2(t)\, dt = \int_0^{500} t\,[r(t) - y(t)]^2 dt \tag{19}$$

where the weightes are assigned as $w_1 = w_2 = w_3 = w_4 = 10$.

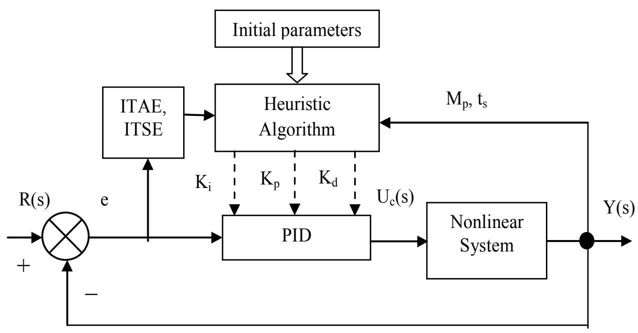

Fig. 4. Soft computing based PID controller tuning.

Prior to the optimization search, it is necessary to assign the parameters for the heuristic algorithm. The initial algorithm parameters considered in the proposed work is presented in Table 2. The algorithm is allowed to explore the search universe until the OF value is minimized. This procedure is repeated 10 times and the average value of the controller parameters is chosen as the best possible controller parameter value.

Table 2. Initial parameters of heuristic algorithms considered.

Parameter	PSO	BFO	FA	CS
Maximum Number of Iterations	500			
Population of agents (*N*)	15			
Search dimension (*D*)	3			
Stopping criteria	J_{min}			
Number of trials	*10*			
Performance measure values	M_p, t_s, *ITAE, ITSE*			

5. Results and Discussions

The heuristic algorithm based controller tuning procedure is demonstrated using a PSO, BFO, FA, and CS algorithm using Hopper tank and Spherical tank process models. In order to perform a fair comparison, all the algorithms are assigned with the same nominal parameters as shown in Table 2.

5.1. Hopper tank model

Initially, the proposed controller design procedure is implemented using the Hopper tank model discussed in Section 3.1. For this tank, Kesavan et al. [12] developed four FOPTD models based on four operating regions.

Initially, the FOPTD model of I region is considered and the controller tuning procedure is executed. The soft computing approach continuously adjusts the values of K_p, K_i, and K_d until the objective function J_{min} reaches a minimal value. During the optimization search, the PSO based approach converges at 49th iteration, BFO based search converges at 73rd iteration, FA based search converges at 55th iteration and CS based search converges at 64th iteration. From these results, it is noted that, PSO offers faster convergence compared to the alternatives considered in this study.

The controller parameters obtained using the soft computing technique is shown in Table 3. The performance of the controller is tested with an unity reference input and an input disturbance of 50% of reference input (0.5) introduced at 10 seconds. The process response and the corresponding controller response are presented in Figs. 5 and 6 respectively. Corresponding performance measure values, such as M_p, t_s, *ITAE* and *ITSE* are presented in Table 3. From this, one can observe that, the proposed method supports smooth reference tracking and effective input disturbance rejection operations.

Table 3. Controller values and performance measure values for I Region.

Method	K_p	K_i	K_d	M_p	t_s (s)	ITAE	ITSE
PSO	0.5098	0.4929	0.0136	0.000	5.418	12.49	3.934
BFO	0.5217	0.7038	0.0424	0.021	4.862	8.525	2.774
FA	0.4486	0.7084	0.0277	0.042	4.426	8.774	3.105
CS	0.5925	0.6792	0.0218	0.000	3.317	8.597	2.666

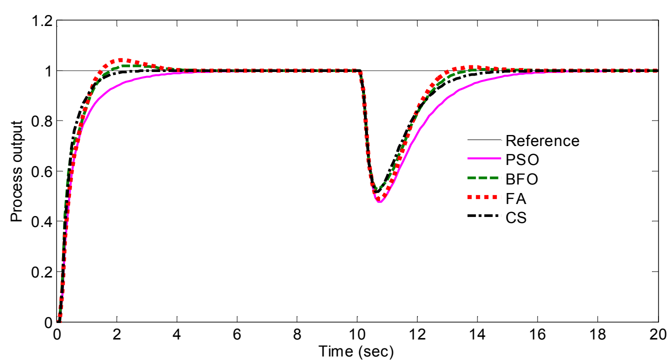

Fig. 5. Process output response with I Region model.

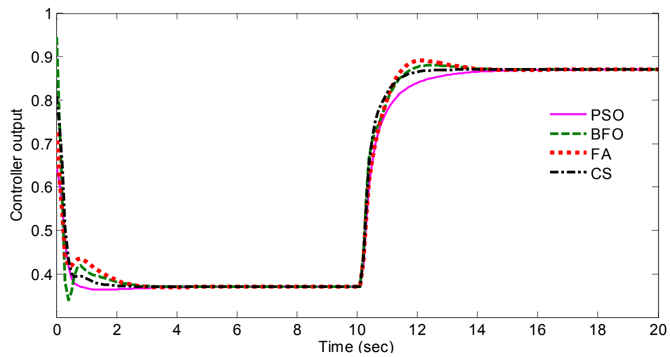

Fig. 6. Controller output response with I Region model.

Similar procedure is repeated for other FOPTD models of the Hopper tank system and the results are clearly presented in Tables 4, 5, and 6 and Figs. 7 to 12. From these results, it is noted that, the BW based BFO, FA, and CS algorithms offers better result compared with the PSO based controllers.

Table 4. Controller values and performance measure values for II Region.

Method	K_p	K_i	K_d	M_p	t_s (s)	ITAE	ITSE
PSO	0.6239	0.4922	0.1083	0.000	9.831	37.18	6.992
BFO	0.6111	0.4306	0.0703	0.000	14.74	43.30	8.105
FA	0.6155	0.4719	0.1104	0.000	10.95	38.90	7.334
CS	0.7204	0.5003	0.0829	0.000	10.27	35.75	6.490

Table 5. Controller values and performance measure values for III Region.

Method	K_p	K_i	K_d	M_p	t_s (s)	ITAE	ITSE
PSO	1.0874	2.2180	0.0937	0.000	11.88	11.05	2.026
BFO	1.6936	2.5921	0.1162	0.000	9.772	8.987	1.460
FA	2.0488	2.5170	0.1091	0.000	10.64	9.795	1.459
CS	2.1083	3.0037	0.0774	0.000	8.830	7.262	1.100

Table 6. Controller values and performance measure values for IV Region.

Method	K_p	K_i	K_d	M_p	t_s (s)	ITAE	ITSE
PSO	5.1824	7.1271	0.0722	0.000	10.02	4.028	0.597
BFO	5.0038	6.1183	0.0320	0.000	10.98	5.352	0.794
FA	4.0273	6.9924	0.0533	0.000	9.135	3.905	0.646
CS	4.9277	8.3081	0.0660	0.000	8.006	2.959	0.458

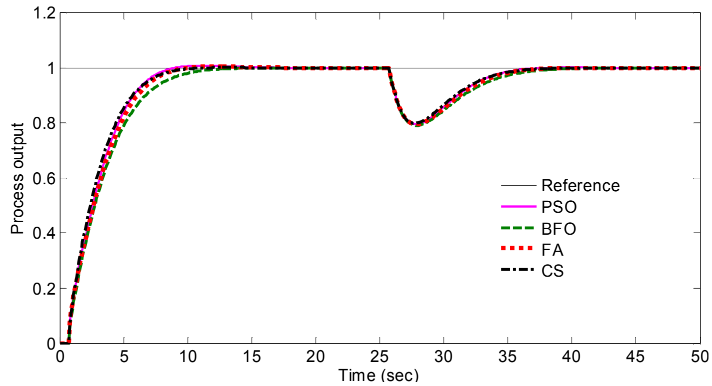

Fig. 7. Process output response with II Region model.

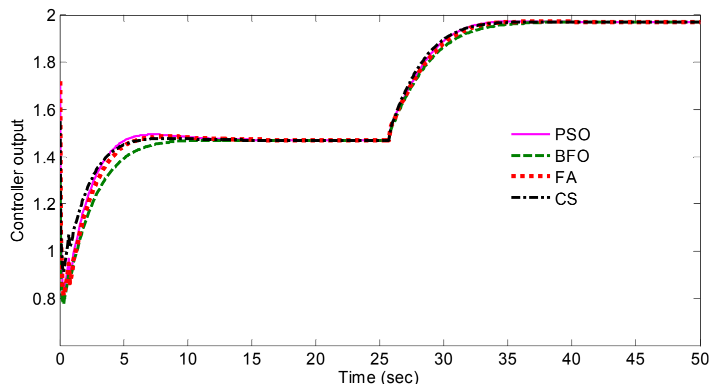

Fig. 8. Controller Output response with II Region model.

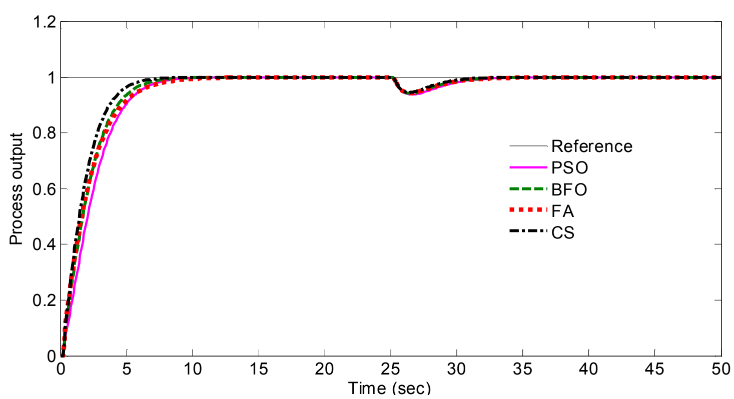

Fig. 9. Process output response with III Region model.

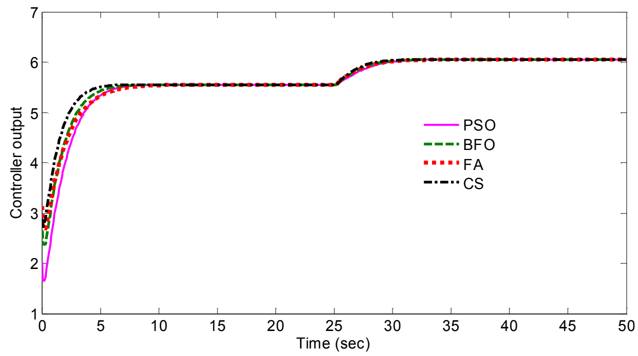

Fig. 10. Controller output response with III Region model.

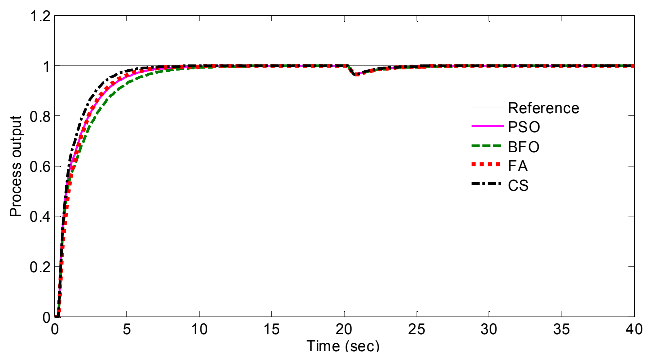

Fig. 11. Process output response with IV Region model.

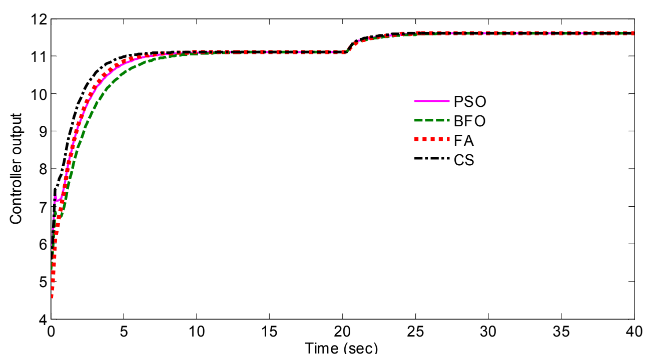

Fig. 12. Controller output response with IV Region model.

Figure 13 presents the comparison of ITAE and ITSE for the Hopper tank system (regions I to IV). From this result, it is observed that, compared with the PSO, the BW guided algorithm offers reduced error values. Particularly, BW guided CS algorithm outperforms the alternatives considered in this study. It offers better ITAE and ITSE values compared with PSO, BFO and FA.

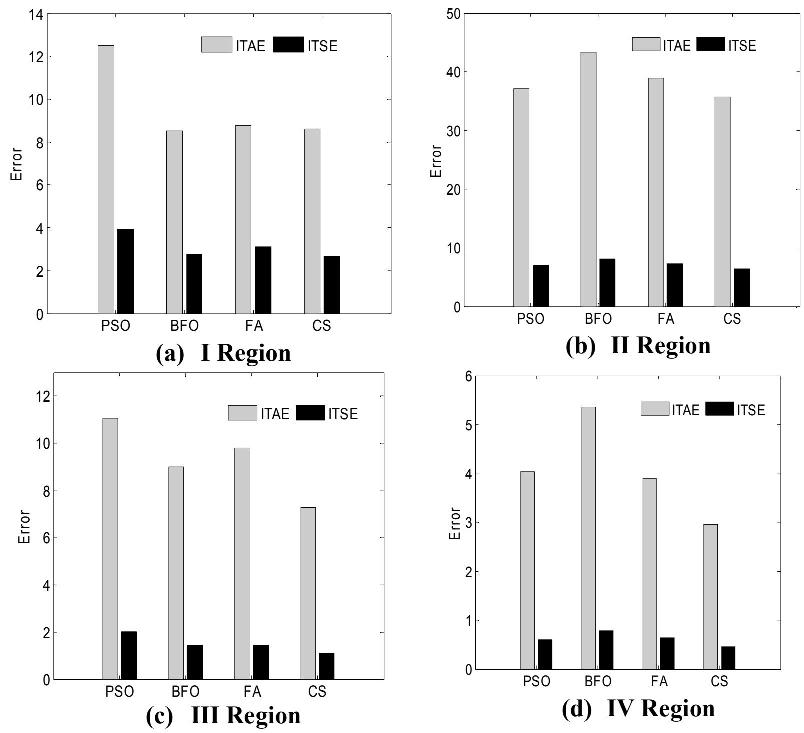

Fig. 13. Error comparison for Hopper type model.

5.2. Spherical tank model

The proposed controller design procedure is then implemented on the FOPTD model of spherical tank system discussed in Section 3.2. During the controller design process, the procedure considered in Section 5.1 is repeated with the spherical tank model. The average search iteration is as follows, PSO converges at 153rd iteration, BFO based search converges at 227th iteration, FA converges at 182nd iteration and CS converges at 194th iteration

Final controller parameters considered in this study is shown in Table 7. The performance of the PID controller for spherical tank system is tested for reference tracking and input disturbance rejection operations. The input disturbance value of 0.5 is introduced at 500 seconds as depicted in Figs. 14 and 15. The process response and the corresponding controller response is presented in Figs. 14 and 15 respectively. Corresponding performance measure values are presented in Table 7. The BW guided FA offers better M_p and t_s compared with the PSO, BFO and CS.

Figure 16 shows the assessment of ITAE and ITSE for the spherical tank system. From this, it is noted that, even though the search iteration is large, BFO results in providing less error value when compared to other methods.

Table 7. Performance measure values for spherical tank system.

Method	K_p	K_i	K_d	M_p	t_s (s)	ITAE	ITSE
PSO	3.4833	0.0926	1.5441	0.4502	208.01	6582	1058
BFO	4.7204	0.1036	1.1943	0.4655	129.47	**3880**	**676.3**
FA	4.2027	0.0805	1.0073	**0.3731**	**102.68**	4924	727.9
CS	3.8837	0.1175	1.1727	0.5438	242.91	5968	1048

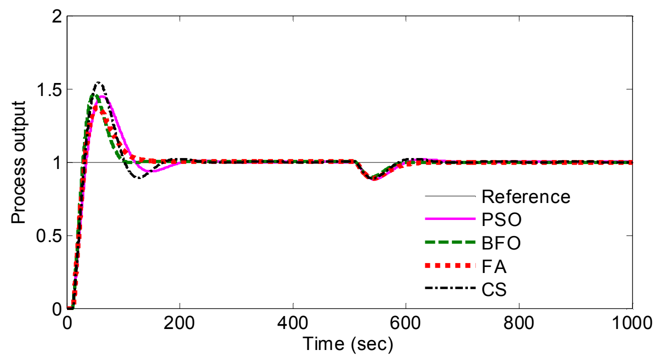

Fig. 14. Process output response of spherical tank process.

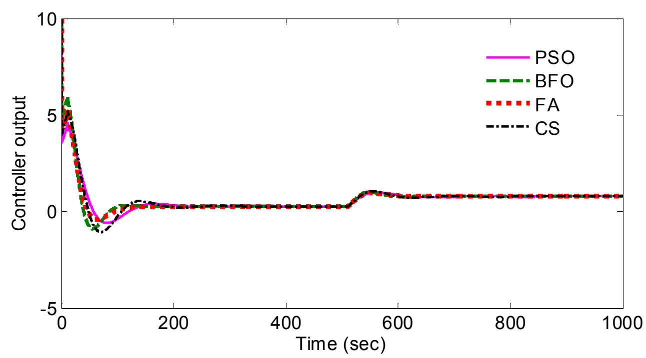

Fig. 15. Controller output response of spherical tank process.

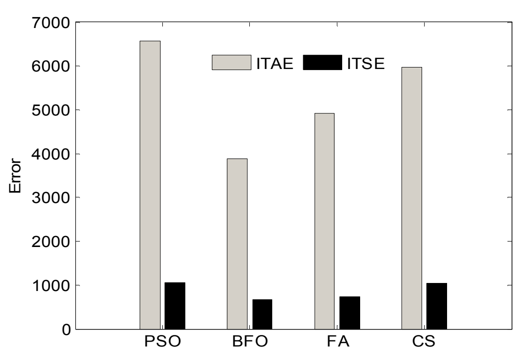

Fig. 16. Error values of spherical tank process.

6. Conclusions

In this paper, PID controller tuning is proposed for hopper type tank and spherical tank system using PSO, BFO, FA and CS. The performance of the proposed procedure is validated using traditional process measures, such as M_p, t_s, *ITAE*, and *ITSE*. The result shows that the PSO-based approach offers a faster convergence when compared with BFO, and proposed BW guided FA and CS. However, even though there is a deviation in the performance measure values, BW guided algorithms offers overall superior result than PSO tuned controller for both the hopper type tank and spherical tank system.

References

1. O'Dwyer, A. (2009). *Handbook of PI and PID controller tuning rules*. (3rd Ed.). Imperial College Press, London.
2. Johnson, M.A.; and Moradi, M.H. (2005). *PID control: new identification and design methods.* (1st Ed.). Springer, London, UK.
3. Vijayan, V.; and Panda, R.C. (2012). Design of a simple setpoint filter for minimizing overshoot for low order processes. *ISA Transactions*, 51(2), 271-276.
4. Vijayan, V.; and Panda, R.C. (2012). Design of PID controllers in double feedback loops for SISO systems with set-point filters, *ISA Transactions*, 51(4), 514-521.
5. Bahita, M.; and Belarbi, K. (2012). Neural stable adaptive control for a class of nonlinear systems without use of a supervisory term in the control law. *Journal of Engineering Science and Technology*, 7(1), 97-118.
6. Besheer, A.H. (2011). Wind driven induction generator regulation using ant system approach to Takagi Sugeno fuzzy PID control. *WSEAS Transactions on Systems and Contro*l, 6(12), 427-439.
7. Manic, S.K.; Sarath, A.; and Rajinikanth, V. (2014). Refined double search optimization methodology to design PID controller for unstable systems. *Australian Journal of Basic and Applied Sciences*, 8(10), 44-51.
8. Liu, G.P.; Yang, J.-B.; and Whidborne, J.F. (2008). *Multiobjective optimization and control*. Printice Hall, New Delhi, India.
9. Kotteeswaran, R.; Sivakumar, L. (2014). Lévy guided firefly algorithm based tuning of decentralised PI controller of nonlinear multivariable system-coal gasifier. *Proceedings of Advances in Control and Optimization of Dynamical Systems IFAC*, 3(1), 127-134.
10. Kotteeswaran, R.; Sivakumar, L. (2014). Performance evaluation of optimal PI controller for ALSTOM gasifier during coal quality variations. *Journal of Process Control*, 24 (1), 27-36.
11. Latha, K.; Rajinikanth,V.; and Surekha, P.M. (2013). PSO-Based PID controller design for a class of stable and unstable systems. *ISRN Artificial Intelligence*, Vol. 2013, Article ID 543607, 11 pages.
12. Kesavan, S.M.; Padmesh, TVN.; and Shyan, C.W. (2014). Controller tuning for nonlinear hopper process tank – a real time analysis. *Journal of*

Engineering Science and Technology, EURECA 2013 Special Issue August, 59-67.

13. Rajinikanth, V.; and Latha, K. (2012). Tuning and retuning of PID controller for unstable systems using evolutionary algorithm. *ISRN Chemical Engineering*, Vol. 2012, Article ID 693545, 11 pages.
14. Rajinikanth, V.; and Latha, K. (2012). Controller parameter optimization for nonlinear systems using enhanced bacteria foraging algorithm. *Applied Computational Intelligence and Soft Computing*, Volume 2012, Article ID 214264, 12 pages.
15. Kennedy, J.; and Eberhart, R.C. (1995). Particle swarm optimization. *In Proceedings of IEEE international conference on neural networks*, 1942-1948.
16. Sri Madhava Raja, N.; Suresh Manic, K.; and Rajinikanth, V. (2013). Firefly algorithm with various randomization parameters: an analysis. *Proceedings of the 4th International Conference on Swarm, Evolutionary, and Memetic Computing* (SEMCCO '13), B. K. Panigrahi, P. N. Suganthan, S. Das, and S. S. Dash, Eds., *Lecture Notes in Computer Science*, Vol. 8297, 110-121.
17. Yang, X.-S. (2008). *Nature-inspired metaheuristic algorithms*. Luniver Press, UK
18. Sri Madhava Raja, N.; Rajinikanth, V.; and Latha, K. (2014). Otsu based optimal multilevel image thresholding using firefly algorithm. *Modelling and Simulation in Engineering*, Vol. 2014, Article ID 794574, 17 pages
19. Yang, X.-S.; and Deb, S. (2009). Cuckoo search via Lévy flights. *Proceeings of World Congress on Nature & Biologically Inspired Computing* (NaBIC 2009), *IEEE Publications*, USA, 210-214.
20. Kotteeswaran, R.; and Sivakumar, L. (2014). Optimal tuning of decentralized PI controller of nonlinear multivariable process using archival based multiobjective particle swarm optimization. *Modelling and Simulation in Engineering*, Vol. 2014, Article ID 504706, 16 pages.